Atari Computer Enthusiasts (N.S.W.)

INSIDE INFO No. 6 April 1983

1ST AUSTRALIAN PERSONAL COMPUTER SHOW
ΞΟΨ ΥΘΕ ΑΥΑΣΙ ΤΠΕΑΛΤ ΗΣΕΕΛ
10FA=4346 00101100=44 EC=1110100
FORTH

Atari Computer Enthusiasts (N.S.W.)

Atari Computer Enthusiasts (N.S.W.) is an independent, non-profit computer users' group loosely affiliated with Atari Computer Enthusiasts in the U.S.A. We have no connections with ATARI, Inc. or their Australian distributors, Futuretronics Australia Pty Ltd.

Our aims include promotion of the ATARI 400/800 Home Computer System; instructing both beginners and advanced users in programming techniques; exchanging hints, tips and ideas amongst members and generally enjoying ourselves.

Meetings are held at 6.0 P.M. on the first Monday of every month (or the second Monday if it clashes with a public holiday) in the Amenities Room, 7th Floor, OTC House, 32-36 Martin Place, Sydney (turn right when you get out of the lift).

Membership to A.C.E.(N.S.W.) is $15 one-time joining fee plus $15 annual subscription (or $10 joining fee plus $10 annual subscription for students under 18 and still at school).

Subscriptions or postal enquiries may be directed to:

Atari Computer Enthusiasts (N.S.W.)
G.P.O. Box 4514,
Sydney,
N.S.W. 2001

Phone enquiries:

Tony Reeve (President): 452-2974
Barry Williams (Vice President): 452-2229
Steven Marcus (Secretary/Treasurer): 387-4287
Garry Francis (Editor): 789-1397

Meeting Dates

*11th April (due to Easter)
2nd May
*6th June
4th July
*1st August
5th September
*10th October (due to 8-Hour Day Public Holiday)
7th November
*5th December

* indicates release dates for INSIDE INFO (subject to unforseen circumstances). Deadline for articles is the meeting prior to the release date.

* * *

<u>WANTED:</u> Computer oriented jokes, riddles, words of wisdom and other silly little tid-bits to help fill empty spaces like this. Or how about some one-liners? Try these for size...

```
1DEG:GR.23:F.I=708TO710:POKEI,255*RND(I):N.I:F.I=0TO359:C.4*RND(I):PL.80,48:DR.80+47*SIN(I),48-47*COS(I):N.I:RUN

2GR.24:X=0:Y=0:SE.2,0,0:C.1:F.I=1TO48:PL.X,Y:DR.X,191-Y:DR.319-X,191-Y:DR.319-X,Y:DR.X,Y:X=X+5:Y=Y+4:N.I:RUN

3GR.23:DEG:F.I=1TO3:R=8+I*12:C.I:PL.79+R,47:F.J=75TO1800STEP75:DR.79+R*COS(J),47+R*SIN(J):N.J:N.I:F.I=1TO9999:N.I
```

From The President

The 1st Australian Personal Computer Show has been hailed a great success for the organisers and exhibitors. The crowds were continuous over the 3 days and interest shown by the general public indicates that this is certainly a growth industry.

The A.C.E.(N.S.W.) stand was very small compared with other exhibitors, however we made up for that with our enthusiasm. This was the first major public relations project undertaken by your club and I am pleased to report that it was very successful in advertising our activities to the public.

On your behalf, I would like to thank Barry Williams for organising our display, Garry Francis for his attendance and help every day, the committee and other members who assisted on the stand. Our friends at Computer Wave provided the computer, so thanks again to Bernhard for his continual support.

- Tony Reeve

* * *

Editorial

Please note the new address for meetings!! We can no longer use I.P. Sharp's premises due to new security arrangements and what not. I'd like to thank them for putting up with us while they did. We couldn't have got established in the early days without their facilities. From now on, meetings will be in OTC House (details page 2). Please respect this privilege and be on your best behaviour at all times, otherwise you'll get hoicked out!

The 1st Australian Personal Computer Show indicated that the ATARI is alive and well and living quite nicely in Australia, thank you very much. There's a full report on the show in this issue. In addition, there's a couple of games, a demo, a Greek alphabet, the solution to Doggies and articles on number systems and the ATARI's extra sound capabilities. We also present our first article on FORTH. I'm very pleased about this one, as there's a lot of members interested in FORTH at the moment. As we are also getting FORTH in the Software Exchange, it seems like an opportune time to begin a FORTH Special Interest Group. Who'd like to get it started?

Last issue, I promised a discussion of ANTIC modes 4 & 5. As it happens, Tim Kilby beat me to it. He has given a very good coverage in the December 1982 issue of BYTE, so it seems pretty pointless to repeat what he has already said. His article also includes an ANTIC mode 4 & 5 character editor. (If you don't get BYTE, there's also an editor in COMPUTE! February 1983.)

I'm extremely pleased with the increase in the number of article submissions recently, but we still need more. How about some utilities and business applications? And what about education? We haven't had a single submission of educational software yet!

This issue ends our first year of publication. As promised in Issue No.1, we are now ready to award $100 worth of software to the author of the best article(s) published in the last 12 months. This is your chance to show that you're truly interested in the magazine by submitting your vote. Simply put the name of the author who you think most deserves the prize into an envelope and send it to The Editor, INSIDE INFO, G.P.O. Box 4514, Sydney, N.S.W. 2001. All votes must be in by the next meeting, i.e. 2nd May, 1983. If you're a recent member and don't have all the back issues, then just base your vote on the issues you have seen. We have now got well over 150 members, so let's see 150 votes!

Next issue is on scrolling. I've received an article on scrolling from one of our New Zealand members. It includes a really excellent demo of horizontal scrolling done properly! I've also got a vertical scrolling game in mind, but no promises on this one. In addition, there will most likely be an assembler written in BASIC and at least one other game. Bye 'til then.

- Garry Francis

1st AUSTRALIAN PERSONAL COMPUTER SHOW

by Garry Francis

The 1st Australian Personal Computer Show began at 7.30 A.M. on Thursday, 10th March with a champagne breakfast in the Green Room at Centrepoint. (No corn flakes for yours truly on this morning!) It was here that the show was officially opened by The Attorney General of N.S.W. The Honourable Mr Paul Landa. Quite surprisingly, the finale was given by a speech synthesizer which sounded unmistakably like S.A.M., the Software Automatic Mouth available for the ATARI. The crowds started rolling in at 9 o'clock and they just didn't stop for 3 days! There was a long queue for tickets for the entire show. Pre show publicity had suggested that attendance would be about 10,000. Actual figures quoted after the show were 35,000!! The show was a huge success for all concerned and it was a terrific feeling to be a part of it. There was an air of interest and enthusiasm everywhere you went. If you missed it...well, I'm just glad I didn't!

We had a stand at Computer Club Corner on the Executive Level (the 3rd floor of the exhibition complex). This was provided to us free courtesy of the organisers. I spent most of the first 2 days manning our stand, but I devoted virtually the entire 3rd day to discussions with various exhibitors. As a result of this, we have come up with a number of great deals - cheap discs, books, software and modems and guest speakers for meetings. Make sure you read every word of future issues of INSIDE INFO in order to avoid missing out on anything!

The major attraction for ATARI owners (or potential owners) was, of course, the Futuretronics stand. I must admit that I was extremely pleased to see them there, as early advertisements for the show gave the impression that they would not be present. They had a very nice display, but it did not have the size or impact of Apple, Commodore, Texas Instruments and others. I was promised all sorts of new product information and prices, but despite a follow-up phone call, it had still not arrived at the time of writing. The following information is from personal observation and research, so it may not be accurate.

The most amazing new product was the WICO Trackball ($89). It is absolutely marvellous. You haven't lived until you've played Centipede with the trackball! They also had the WICO joysticks ($34 and $40) and ATARI's remote control joysticks ($115).

ATARI 400 owners will be interested in a new full stroke keyboard which sits directly in the computer to replace the old membrane keyboard. A Futuretronics salesman said that this was developed by their own research and development team...I didn't know they had one. Price not known at present.

New ATARI software includes Galaxian and Defender (about $89). A plethora of new ATARI software has been released (or announced) in the U.S.A., but is not available here just yet. Of particular interest is an excellent new ROM based word processor. New games include Qix, Dig Dug, ET Phone Home, Superman III, Star Trux and Donkey Kong. Educational programs include Juggles' Rainbow and Juggles' House. There is also some interesting releases from APX including Galahad and the Holy Grail, Salmon Run, Typo Attack (a typing tutor in the form of a game), Quarxon, Real Estate Cash Flow Analysis, BASIC XA (an extended ATARI BASIC) and lots more. Availability and prices for these items are not yet known.

Futuretronics also distributes selected products from Datasoft and Synapse. New products from Datasoft include the long awaited BASIC compiler ($207.95), although at this price, I think the new compiler from Monarch Data Systems, P.O. Box 207, Cochituate, MA 01778, U.S.A. will be more popular. It is $69.95 in the U.S.A. Zaxxon has been advertised in the U.S.A. and we hope it gets here soon.

Futuretronics also had Astro Chase, the amazing new game written by Fernando Herrera for his new company called First Star Software. It's a space game with multi-directional scrolling somewhat similar to the arcade game Bosconian.

I've left the best 'til last. Yes, the new ATARI 1200XL was also on display. It is not yet known when it will come to Australia (or if in fact it even comes to Australia), but it's worth mentioning. I had a chance to have a good long play with it towards the end of the show and I must say it looks and feels great. The keyboard layout is much the same as the 400/800 except that the BREAK, Inverse Video, SYSTEM RESET and other console keys are moved to a row at the back of the keyboard in addition to 4 function keys (F1 to F4) and a HELP key for carrying out system diagnostics. The function keys may be used to move the cursor around with one keystroke. SHIFT plus a function key sends the cursor to the respective corners of the screen display. CONTROL plus a function key disables the keyboard, DMA or keyboard click (F1 to F3 respectively) or toggles between the familiar ATARI character set and an international set (F4). It has an enhanced 14k Operating System which allows redefinition of the entire keyboard. You can now directly access ANTIC modes 4, 5, 12 and 14 (the latter are often referred to as GRAPHICS 6+ and 7+), by using GRAPHICS 12 to 15 respectively. This is something that I believe should have been in the original Operating System right from the start.

Now for the bad news. The 1200XL comes with 64k RAM as standard and is completely sealed (no more RAMdiscs or 80 column boards). The Operating System loads from ROM into RAM, hence there is no usable increase in RAM when using BASIC, etc. The Operating System may be disabled to give you 62k of RAM, but you are left entirely on your own. No CIO! No vertical blank interrupt service routines! No device handlers! No floating point package! Nothing! There's only one cartridge slot (no more Monkey Wrenches) and only 2 joystick ports (no more peripheral interfaces through Ports 3 and 4). The 6502 is supposedly a custom made job. The sound is supposed to be cleaner and the colours are supposed to be 100% more saturated with no more colour bleeding, but it all looked and sounded the same to me. Overall, a very nice computer if it had come out 4 years ago (before the 400 and 800), but limited by its inability to be expanded. There have been several full reviews published, so for more details see ANTIC February/March 1983 (written by an ATARI employee and hence very favourable), M.A.C.E. Journal January 1983 and COMPUTE! March 1983.

Also big news from Futuretronics was that they are ready to expand the ATARI in Schools Project (ASP) into N.S.W. If their tender is successful, they would offer ATARI computer packages to schools at ridiculously low prices thanks to financial assistance from ATARI themselves. I wish them luck in this venture, as the ATARI 400/800 is undeniably the best computer available for school use. Apart from which, when parents are later persuaded to buy a home computer, they will naturally be swayed by the fact that little Johnny or Belinda uses an ATARI at school. They would then be buying the best home computer on the market.

The most exciting attraction at the show was no doubt the presence of Winfried Hofacker representing his company Elcomp Publishing Inc. He had travelled all the way from Germany especially for the show. Elcomp carries a tantalising range of books, software and hardware projects for the ATARI 400/800. They had 2 books which have been available overseas for some time, but were virtually unavailable here. These are "ATARI BASIC - Learning by Using" and "Games for the ATARI". Also available were a number of new books, viz. "How to Program Your ATARI in 6502 Machine Language", "FORTH on the ATARI - Learning by Using" and "Astrology - Use Your ATARI Computer to See the Future". The books on FORTH and machine language were especially popular amongst those manning our stand (I think everybody bought the book on FORTH). Another very interesting book called "Hackerbook 1" will be available around May. This is full of hints and tips on all the more difficult topics for intermediate to advanced programmers. Also coming at a later date is a book on applications in FORTH.

Their software is also quite interesting. They include discs of programs from the BASIC and Astrology books, a word processor, machine language monitor, tracer/debugger, macro assembler, moon phases, biorhythms, birth control and a number of business programs for mailing lists, inventory, invoicing, appointments and a very interesting one which converts your ATARI into a cash register. A popular program was Elcomp's version of FORTH. This is an extended fig-FORTH with editor, I/O package, joystick control, graphics, sound, player-missile graphics and more. An extra disc is available with a floating point package and trig functions. Or if you're just a beginner, you may like to consider their learn-FORTH, which is a subset of fig-FORTH and is also available on cassette.

For the hardware buff, there is a number of interesting add-ons, usually in the form of kits with full construction details and software where applicable. These include an Epson printer interface which uses Ports 3 and 4 instead of the 850 interface module, an RS232 interface, EPROM burner and do-it-yourself cartridges.

I met a lot of nice people at the show, but Winfried would have to take the cake. He is bubbling with enthusiasm for his products and, unlike many others in the industry, he is not a crook. Everything is reasonably priced. He is presently looking for distributors for his products in Australia and hopes to sell them at the same price as in the U.S.A. A product information sheet is enclosed with this issue to give you an idea of prices. He has agreed to sell bulk quantities to the club at dealer prices, so let us know if you're interested. We will not be persuing this (or any of the other many offers mentioned in this report) unless sufficient people respond! So if you don't get off your butt and say something, nothing will happen.

Wizware is a very interesting new range of educational software being released by Ashton Scholastic Pty. Ltd., P.O. Box 579, Gosford, 2250. The first releases on 15th May will be The Match Game and Turtle Trax on cassette for $29.95 each and The Microzine and Nerd Alert on disc for $39.95 each. They did not have the ATARI programs on display, but the Apple and VIC-20 versions looked quite good. They are also expecting Bank Street Writer, the famous word processor developed for children, yet powerful enough that adults are clamouring for it. (See the review in TIME magazine 14th March, 1983.) It sells for $69.95 in the U.S.A. and should be similarly priced here.

At long last, the excellent range of Thorn EMI software is ready to be released in Australia. They are presently looking for a distributor, but if all goes well, the programs will be available at the end of April. Prices are not yet available. If you weren't at the show, you might like to contact Thorn EMI, 301 Castlereagh Street, Sydney, 2000 and ask for their catalogue. There are currently 4 cartridges and 15 cassettes. The cartridges are Kickback, Soccer, Submarine Commander and Jumbo Jet Pilot. The latter 3 are superb. The cassettes are for Home Financial Management, Darts, Snooker and Billiards, Pool, Dominoes and Cribbage, Humpty Dumpty and Jack and Jill, Hickory Dickory Dock and Baa Baa Black Sheep, British Heritage Jigsaw Puzzles Vol. 1 & 2, European Scene Jigsaw Puzzles Vol. 1 & 2, Supercubes and Tilt, Figure Fun, Compute 4 and Reversi, Owari and Bull and Cow. We will have full reviews of these programs in future issues and demonstrations at future meetings. There is also more to come in the future. One of the more interesting projects will be the co-release of the movie "War Games" and the computer game of the same name. I am told that it is about a boy who uses his ATARI to tap into the Pentagon computer and cause all sorts of havoc.

At the time of writing, Dick Smith Electronics is selling a pair of ATARI paddles for $21 (normally $39.95). You'd be mad not to buy a pair at this price. He's also selling the Dataphone direct connect modem for $199. ATARI compatability doesn't seem to be a problem.

Talking about modems, Electro-Med now has a version of the Sendata 700 series answer/originate acoustic coupler especially for the ATARI. They offered us a very good deal on bulk orders. The normal retail price is $298.80, but we can buy it at the retailers' price of $209.16 to $239.04 depending on quantity. As we are looking into a Bulletin Board Service and computer communications is rapidly becoming more popular, now is the time to start thinking about a modem. Please let us know if you're interested. We need to buy 21 to get the cheapest price.

Imagineering did not have any ATARI software on display, however I did get invited to contact them after the show. They have a wide range of software from companies such as Sierra On-Line, Broderbund, Roklan, SSI, Sirius, LJK Enterprises, Visicorp, Eduware, PDI, Infocom, JV Software, Quality Software, Gebelli, etc. etc. etc. If you see a program you want in an overseas magazine, then ask your dealer to get it in. If it's recent or popular, there's a good chance Imagineering will have it. A couple of new ones of interest are Gorf and Ultima II. I am told there is a lot of new stuff coming soon. They also distribute the indispensible "Your ATARI Computer" by Poole, McNiff and Cook and "The Book of ATARI Software 1983" edited by Stanton, Wells and Rochowansky. I've ripped the price tags off mine, but I think they were $23.75 and $29.95 respectively.

For any out of towners, you can now order all the Imagineering software from a Newcastle-based mail order house called Micro-Educational, 17 Park Road, Garden Suburb, 2288. Phone (049) 43-6805. Ring or write and ask for their catalogue.

A rep from Videoactiv informs us that they will be carrying a range of cartridges from Spectravision. The first one is called Goldmine and will retail for about $45. Others are waiting to be shipped from Hong Kong. Spectravision is already known for their Quik-Shot joystick ($24.95). They also have a range of VCS cartridges. For further details, cotact Tony King at Videoactiv, G.P.O. Box 4446, Sydney, 2001 or phone 909-2449.

Books. Books. Books. I never dreamed there could be so many new ATARI books. There's honestly something for everyone.

The Australia and New Zealand Book Co. had "BASIC Exercises for the ATARI" by Lamoitier (SYBEX) for $21.50.

Doubleday Australia will be carrying a number of new titles. From Creative Computing Press comes "The Creative ATARI" by Small, Small and Blank $15.95 and "Computers for Kids (ATARI Edition)" by Sally Greenwood Larson $4.95. From ARCsoft comes "ATARI Computer Program Writing Workbook" by Alan North $4.95, "101 ATARI Computer Programming Tips and Tricks" by Alan North $8.95 and "31 New ATARI Computer Programs for Home, School and Office" again by Alan North $8.95. There are 3 interesting titles in the series of Handy Guides from Alfred Publishing at $3.95 each. These are "How to Use the ATARI Computers" by Michael Boom, "Understanding FORTH" by Joseph Reymann and "ATARI Graphics" by Michael Boom. Note that all the Doubleday prices are U.S. prices, so you can expect to pay a couple of dollars more in Australia.

The Technical Bookshop had its usual tantalising range of computer books including "Picture This" for PILOT users by David Thornburg $14.95. I also grabbed their Osborne/McGraw-Hill catalogue which had a very interesting book called "ATARI 400/800 Disk Guide" by John Taylor. Availability and price is not yet known.

One new book that I didn't see anywhere was "Inside ATARI BASIC" by ATARI's Sales Training Director Bill Carris. It's published by Reston.

We were approached by Prentice-Hall of Australia, who had "The VisiCalc Book: ATARI Edition" by Beil $23.25, "ATARI PILOT for Beginners" by Conlan $23.25 and "FORTH Programming" by Scanlon $26.25 as well as a number of old titles.

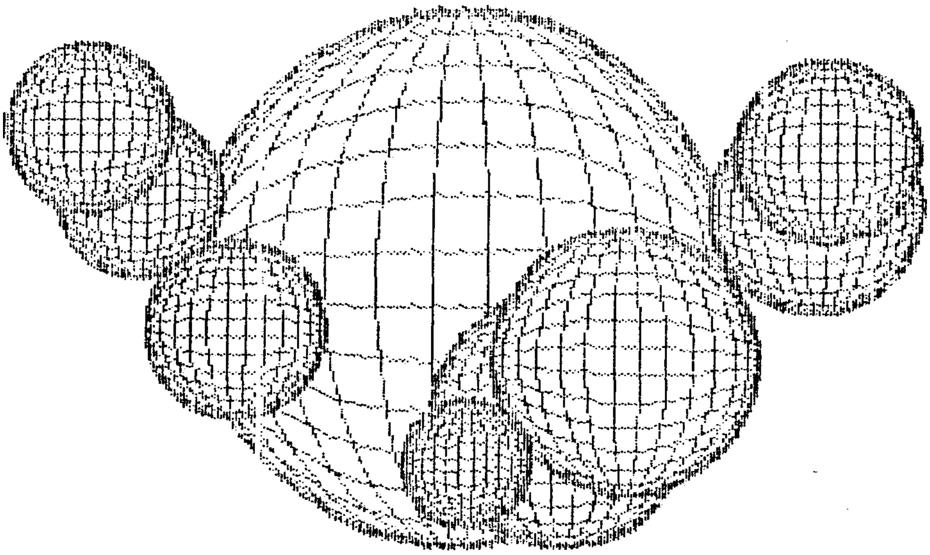
We were also approached by a company who will be distributing COMPUTE! Books. They have several books of interest to ATARI owners, including the well known "COMPUTE!'s First Book of ATARI", "COMPUTE!'s Second Book of ATARI" and "Inside ATARI DOS" by Bill Wilkinson. I have got all 3 and the latter 2 in particular are excellent. 2 new titles are "COMPUTE!'s First Book of ATARI Graphics" and "Mapping the ATARI". I saw the latter book and it is superb! Another book announced in COMPUTE! is "COMPUTE!'s First Book of ATARI Games". The rep hadn't heard of this one yet, but they will no doubt carry it as well. All books are expected to retail for about $24 except for "Mapping..." at $28 and "...DOS" at $36. We hope to be getting these books direct so that members can get them at discount prices. Again let us know if you're interested as we need to get an idea of quantities.

We'll have reviews of lots of these books in future issues.

Well, that just about wraps it up. I'd like to offer a really big thankyou to all those who helped out on our stand, but especially to Computer Wave for lending us a computer, Peter Bamford, Jeff Maddock, Tony Dinallo, Tony Reeve and Jamie Athas who each spent at least one full day on the stand and Barry Williams who not only devoted his time and his TV set, but also organised our stand. Great job Barry! I'd also like to thank Graeme Selby of Australian Exhibition Services and the sponsors - Australian Personal Computer magazine - for making it all possible.

We'll be there again next year, only bigger and stronger, so set aside 15th - 17th March 1984 for the 2nd Australian Personal Computer Show.

* * *



SOLUTION TO DOGGIES

by Garry Francis

Here is the solution to Stan Ockers' Doggies which was published in the last issue. Don't look at the solution until you've had a good go yourself or you'll take all the fun out of it!

Assume the possible positions of the doggies are numbered from 1 (on the left) to 7 (on the right). The starting configuration would therefore be white doggies in positions 1-3, a blank space at position 4 and brown doggies in positions 5-7. There are 2 possible solutions, each being a "mirror image" of the other. One possible solution is given below. See if you can find the other one.

Move
No.
1: Brown from 5 to 4
2: White from 3 to 5
3: White from 2 to 3
4: Brown from 4 to 2
5: Brown from 6 to 4

Move
No.
6: Brown from 7 to 6
7: White from 5 to 7
8: White from 3 to 5
9: White from 1 to 3
10: Brown from 2 to 1

Move
No.
11: Brown from 4 to 2
12: Brown from 6 to 4
13: White from 5 to 6
14: White from 3 to 5
15: Brown from 4 to 3

SOUPED UP SOUND

by J. S. Masters, Howick, New Zealand

Most people are amazed when they first realise that the ATARI has 4 voice sound and all those weird distortions, but it is capable of far more! There's also filters (harmonic and normal), joint voices, variable clocking frequencies, etc. In order to harness all this from BASIC, POKEs have to be used instead of the SOUND command.

The hardware registers from 53760 [$D200] to 53767 [$D207] are the normal volume, pitch and distortion for each voice. The even ones (53760, 53762,...) are POKEd with the pitch. The odd ones are POKEd with the distortion*16+volume. For example, POKE 53760,121:POKE 53761,168 is equivalent to SOUND 0,121,10,8. The hardware register at 53768 [$D208] is called the AUDio mode ConTroL register (AUDCTL) and does numerous things. It may be POKEd with a number from 0 to 255 (where 0 is normal) in accordance with the following bit allocations:

- Bit 0 changes the normal clock frequency from 64kHz to 15khz. This has the effect of greatly dropping the pitch.
- Bit 1 puts a high-pass filter on channel 2. It is clocked by the channel 4 divider. Only sounds higher than channel 4 will be heard.
- Bit 2 puts a high-pass filter on channel 1. It is clocked by the channel 3 divider.
- Bit 3 clocks channel 4 with channel 3. This enables 16 bit resolution, therefore a pitch may be from 0 to 65536 (i.e. 256*256).
- Bit 4 clocks channel 2 with channel 1.
- Bit 5 clocks channel 3 with 1.79 MHz. This increases the frequency by an immense amount.
- Bit 6 clocks channel 1 with 1.79 MHz.
- Bit 7 changes the 17 bit poly counter to 9 bit. This has the effect of increasing the repetitions of the distortions.

Example 1:

```
POKE 53768,30:REM Set bits 1-4
POKE 53760,54:POKE 53761,136:REM SOUND 0,54,8,8. Also 53761,168 sounds good.
POKE 53762,254
POKE 53764,254
POKE 53766,220
```

Example 2:

```
POKE 53768,254:REM Set all bits except 0
POKE 53760,254:POKE 53761,104
POKE 53762,254
POKE 53764,254
POKE 53766,220:REM Affects the speed
```

Another POKE that can be used is 53775 [$D20F]. This is normally 3, but if it is POKEd with 255 it turns on a harmonic filter. This can really give weird and wonderful music when used properly and with the right notes. Voice 0 is the actual filter. It picks harmonics out of voice 1 and provides the volume and distortion (pure tones are best). Voice 1 provides the pitch. Voice 2 and 3 are not used.

Example 3:

```
10 POKE 53762,243:POKE 53761,168
20 POKE 53775,255
30 FOR I=0 TO 255 STEP 0.05:POKE 53760,I:NEXT I
```

For further information, see "De Re ATARI" Chapter 10 (or "The ATARI Tutorial: Part 7" in BYTE, March 1982) and the Hardware Manual. [Both are in the reference library - GF.]

GRAB AN APPLE

by Jamie Athas and Garry Francis

This game is designed in ANTIC mode 4, a character mode which allows up to 4 colours to be displayed within the same character. Each character is set up in a 4 by 8 grid giving a total screen resolution of 160 across by 192 down. To use ANTIC mode 4, it is basically a matter of changing a GRAPHICS 0 display list, then redefining the character set. You are welcome to experiment with the techniques we have used. The game was originally in GRAPHICS 5, but ANTIC mode 4 increases its visual impact tenfold.

How To Play Grab An Apple

This is a game where you control a giant lengthening caterpillar, and have to eat the apple. The quicker you eat the apple, the more points you get, but if you take too long, it will disappear to a new location and you will lose five points. If you run over yourself or hit the wall, you lose a life. You have 3 lives per game. Every time you eat an apple, the caterpillar is "cut up" and your speed increases by 1 km/h.

Program Notes

The concept behind this game is a familiar one, but don't let that fool you. This game is totally original.

When the program begins, a title page is scrolled up the screen. This is for visual effect only and serves no useful purpose, however anybody wishing to examine the code will find it quite interesting. Firstly, let me assure you that it is not genuine scrolling, but display list manipulation. It works by writing your own display list (lines 1010 to 1030) consisting of a screenful of blank mode lines and a few GRAPHICS 2 lines which begin at the bottom of the screen. (This is cheating the system and I'd strongly suggest you avoid such things unless you know what you're doing.) The number of blank mode lines is gradually reduced using the simple FOR...NEXT loop in line 1060, so that the GRAPHICS 2 lines slowly creep towards the top of the screen. The technique is crude, but it works. You'll notice a certain degree of jerkiness, as the movements are not synchronised with the vertical blank. You can experiment with the speed by changing the delay (i.e. FOR W=1 TO 15:NEXT W) in line 1060.

When the scrolling is finished, the character set is copied from ROM to RAM (line 1070) using the machine language routine out of Doggies, then 7 of the special characters are redefined (line 1100). I was going to draw up some pretty pictures to show how this is done in ANTIC mode 4, but unfortunately I ran out of time.

The screen is then cleared and another custom display list is created (line 1160), this time for a screenful of ANTIC mode 4 with 3 lines of GRAPHICS 0 at the bottom for scoring information. If you want an entire screen of ANTIC mode 4 in your own programs, then change the loop to read FOR I=DL+6 TO DL+28. This completes the initialisation.

If you care to draw a flowchart of the main program (lines 20 to 370), you'll notice that it follows a very efficient and logical flow. Beginners should take particular note of this. It is something you should always strive for to achieve peak efficiency without losing readability. This is something I can never stress too much. I get far too many program submissions which have a spaghetti-like structure! As a result, they run slow and use up about twice the memory necessary. In Grab An Apple, the effort has paid off. Speed is no longer a problem despite the delays for sounds. In fact, I had to insert a lengthy delay loop (line 240) to slow the game down. The game becomes more difficult as it progresses by reducing the length of this loop every time you eat an apple. You'll also notice a certain "beat" to the program, which starts the adrenalin flowing as the pace increases. (This was the secret to Space Invaders' success.) You have to eat 50 apples in one game to reach the fastest speed. I certainly can't do it! If you want to see the game at maximum speed, change the loop to read FOR I=1 TO 1:NEXT I.

Anyway, I hope you enjoy playing Grab An Apple as much as Jamie and I did writing it.

```
1 REM ################################
2 REM #        GRAB AN APPLE         #
3 REM # Jamie Athas & Garry Francis  #
4 REM # Published by Atari Computer  #
5 REM #    Enthusiasts (N.S.W.)      #
6 REM #         April 1983           #
7 REM ################################
10 GOSUB 1000
20 LIVES=3:SPEED=1:SCORE=0
30 ? CHR$(125):COLOR 35:PLOT 38,0:DRAW
TO 38,20:DRAWTO 1,20:DRAWTO 1,0:DRAWTO
 39,0
40 HX=4:HY=3:DX=1:DY=0:HEAD=166
50 POSITION 4,21:? "LIVES: ";LIVES:POS
ITION 27,21:? "COUNTDOWN"
60 POSITION 4,22:? "SPEED: ";SPEED;" k
m/h"
70 POSITION 4,23:? "SCORE: ";:COLOR HE
AD:PLOT HX,HY
80 POSITION 11,23:? SCORE;" ";:COUNT=5
0
90 COLOR 32:FOR I=1 TO 2
100 Z=INT(36*RND(I))+2:IF Z=HX THEN 10
0
110 PLOT Z,2:DRAWTO Z,19:NEXT I:FOR I=
1 TO 2
120 Z=INT(18*RND(I))+2:IF Z=HY THEN 12
0
130 PLOT 2,Z:DRAWTO 37,Z:NEXT I
140 AX=INT(36*RND(I))+2:AY=INT(18*RND(
I))+2
150 LOCATE AX,AY,Z:IF Z<>32 THEN 140
160 COLOR 34:PLOT AX,AY:FOR I=10 TO 0
STEP -0.5:SOUND 0,100,10,I:NEXT I:SOUN
D 0,0,0,0
170 POSITION 31,22:? COUNT;" ":BX=HX:B
Y=HY:ST=STICK(0):IF ST=15 THEN 220
180 POKE 77,0:IF ST=14 THEN DX=0:DY=-1
:HEAD=164
190 IF ST=13 THEN DX=0:DY=1:HEAD=165
200 IF ST=7 THEN DX=1:DY=0:HEAD=166
210 IF ST=11 THEN DX=-1:DY=0:HEAD=167
220 HX=HX+DX:HY=HY+DY:LOCATE HX,HY,Z:S
OUND 0,100,12,4:COLOR 161:PLOT BX,BY:C
OLOR HEAD:PLOT HX,HY:SOUND 0,0,0,0
230 IF Z<>32 THEN 260
240 COUNT=COUNT-1:IF COUNT THEN FOR I=
1 TO 51-SPEED:NEXT I:GOTO 170
250 COLOR 32:PLOT AX,AY:SCORE=SCORE-5:
GOTO 80
260 IF Z<>34 THEN 300
270 FOR I=1 TO 50:SOUND 0,I,8,8:NEXT I
:SOUND 0,0,0,0
280 SCORE=SCORE+COUNT:SPEED=SPEED+1:IF
 SPEED>50 THEN SPEED=50
290 POSITION 11,22:? SPEED;" km/h":GOT
O 80
300 FOR I=5 TO 255 STEP 10:SOUND 0,I,1
0,8:NEXT I:SOUND 0,0,0,0
310 LIVES=LIVES-1:IF LIVES THEN 30
320 POSITION 0,21:? CHR$(156);CHR$(156
);CHR$(156):IF SCORE>HI THEN HI=SCORE
330 POSITION 11,21:? "YOUR SCORE: ";SC
ORE:POSITION 11,22:? "HIGH SCORE: ";HI
340 ? "    DO YOU WANT ANOTHER GAME (Y
/N)?";
350 GET #1,Z:IF Z=ASC("Y") THEN 20
360 IF Z<>ASC("N") THEN 350
370 CLOSE #1:POKE 82,2:POKE 106,START+
8:GRAPHICS 0:END
1000 START=PEEK(106)-8:POKE 106,START-
1
1010 GRAPHICS 4:POKE 559,0:POKE 708,70
:POKE 709,108:POKE 710,14:POKE 711,54:
DL=PEEK(560)+256*PEEK(561)
1020 FOR I=DL+3 TO DL+26:POKE I,112:NE
XT I:POKE DL+27,71:POKE DL+28,PEEK(88)
:POKE DL+29,PEEK(89)
1030 FOR I=DL+30 TO DL+33:POKE I,7:NEX
T I:POKE DL+34,65:POKE DL+35,PEEK(560)
:POKE DL+36,PEEK(561):POKE 87,2
1040 POSITION 4,0:? #6;"JAMIE ATHAS":P
OSITION 1,1:? #6;"and garry francis"
1050 POSITION 6,2:? #6;"PRESENT":POSIT
ION 3,4:? #6;"grab an apple":POKE 559,
34
1060 FOR I=96 TO 0 STEP -16:FOR J=DL+3
 TO DL+26:POKE J,I:FOR W=1 TO 15:NEXT
W:NEXT J:NEXT I
1070 DIM ML$(32):FOR I=1 TO 32:READ A:
ML$(I)=CHR$(A):NEXT I:CHSET=256*START:
X=USR(ADR(ML$),57344,CHSET)
1080 DATA 104,104,133,204,104,133,203,
104,133,206,104,133,205,162,4
1090 DATA 160,0,177,203,145,205,136,20
8,249,230,204,230,206,202,208,240,96
1100 FOR I=CHSET+8 TO CHSET+63:READ A:
POKE I,A:NEXT I
1110 DATA 60,255,252,63,255,255,255,60
,10,8,20,85,85,85,85,20,239,239,239,17
0,254,254,254,170
1120 DATA 60,60,190,190,255,255,255,25
5,255,255,255,255,190,190,60,60
1130 DATA 248,248,255,255,255,255,248,
248,47,47,255,255,255,255,47,47
1140 GRAPHICS 0:POKE 559,0:POKE 16,64:
POKE 53774,64:POKE 82,0:POKE 83,39:POK
E 752,1
1150 POKE 708,54:POKE 709,188:POKE 710
,246:POKE 711,90:POKE 756,START
1160 DL=PEEK(560)+256*PEEK(561):POKE D
L+3,68:FOR I=DL+6 TO DL+25:POKE I,4:NE
XT I
1170 OPEN #1,4,0,"K:":HI=0:POKE 559,34
:RETURN
```

NUMBER SYSTEMS

by Brenton Vettoretti

Welcome to the mysteries of number systems. In this article I will attempt to explain binary and hexadecimal number systems. Both these systems should be understood to properly implement redefined character sets, player-missile graphics, modified display lists and so on.

The number system we use in everyday life is the decimal system, which has a base of 10 and uses the numeric characters 0 to 9. It uses positional significance to represent tenths, units, tens, hundreds, etc. For example, the number 2893.45 may be written as:

$$(2x10^{3}) + (8x10^{2}) + (9x10^{1}) + (3x10^{0}) + (4x10^{-1}) + (5x10^{-2})$$

The number system the ATARI uses in its everyday life is the binary system which has a base of 2 and uses the numeric characters 0 and 1. It uses positional significance to represent quarters, halves, units, twos, fours, etc. For example, the binary number 1010.01 may be written as:

$$(1x2^{3}) + (0x2^{2}) + (1x2^{1}) + (0x2^{0}) + (0x2^{-1}) + (1x2^{-2}) = 10.25 \text{ decimal}$$

Hexadecimal numbers (often abbreviated to hex) are used as a shorthand representation of their equivalent binary form. The system has a base of 16 and uses the numeric characters 0 to 9 as well as the alphabetic characters A to F to represent the numbers 10 to 15 respectively. It uses positional significance to represent sixteenths, units, sixteens, thirty-twos, etc. For example, the number A9.C may be written as:

$$(Ax16^{1}) + (9x16^{0}) + (Cx16^{-1}) = 169.75 \text{ decimal}$$

The table below shows binary and hexadecimal numbers with their decimal equivalent.

DECIMAL	HEXADECIMAL	BINARY
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

To convert a number to its decimal equivalent, the most significant character is replaced by its decimal equivalent and multiplied by the base. This is repeated for the next significant character and added to the previous result. This process is continued until the least significant character has been added. The final result is the required decimal number.

Confused? I think everyone is the first time. Try to read from the beginning again, then work through the following examples. Be patient, as this is really logical when you can follow it. Unfortunately, it is very hard to learn to count again in a different base when you have been taught base 10 from your first day at school. Remember the first time you tried to program in BASIC? As with BASIC, you will find a knowledge of other number systems to be extremely useful. On with the examples...

First we will convert the binary number 101100 to its decimal equivalent:

```
                    1  0  1  1  0  0
           (start) x2
                    2 +0
                       2
                      x2
                       4 +1
                          5
                         x2
                         10 +1
                            11
                            x2
                            22 +0
                               22
                               x2
                               44 +0
           (finish)                44
```

i.e. 101100 (binary) = 44 (decimal)

Next we will convert the hexadecimal number 10FA to its decimal equivalent:

```
                    1    0    F    A
           (start) x16        replace with
                    16   +0   decimal
                         16   equivalent
                        x16
                        256   +15
                              271
                              x16
                             4336  +10
           (finish)                4346
```

i.e. 10FA (hexadecimal) = 4346 (decimal)

If you followed the simple arithmetic above, I am sure you will see how easy it is to convert from a strange base to decimal. It is equally simple to convert from decimal to a different base. The decimal number is divided by the base to form a quotient and a remainder. The quotient is then divided by the base. This is repeated until the quotient is zero. The remainders formed by the conversion represent the number in the new base. The last remainder is the most significant digit of the new number. Once again, if you re-read the text and work through the following examples, you will see how simple this is.

First we will convert 236 decimal to binary:

```
                          Quotient  Remainder
           (start) 236/2 =  118         0    (least significant digit)
                   118/2 =   59         0
                    59/2 =   29         1
                    29/2 =   14         1
                    14/2 =    7         0
                     7/2 =    3         1
                     3/2 =    1         1
           (finish)  1/2 =    0         1    (most significant digit)
```

i.e. 236 (decimal) = 11101100 (binary)

Now we will convert 236 decimal to hexadecimal:

```
                          Quotient  Remainder
           (start) 236/16 =  14        12 replace with hex equivalent C (least significant digit)
           (finish) 14/16 =   0        14 replace with hex equivalent E (most significant digit)
```

i.e. 236 (decimal) = EC (hexadecimal)

One further example to convert 62345 decimal to hexadecimal:

		Quotient	Remainder	
(start)	62345/16 =	3896	9	(least significant digit)
	3896/16 =	243	8	
	243/16 =	15	3	
(finish)	15/16 =	0	15 replace with hex equivalent F	(most significant digit)

i.e. 62345 (decimal) = F389 (hexadecimal)

As stated earlier, hexadecimal numbers are a shorthand representation of binary. This can be seen if we split the binary number into groups of 4. Zeroes are added if necessary to complete a group of 4. Each group may then be converted directly to hexadecimal. For example:

Binary	1110	1100	= 236 decimal
Hexadecimal	E	C	= 236 decimal

This article is really only an introduction to different number systems. The techniques shown will allow you to convert from any base to any base. To keep things simple in the early days, I suggest you use decimal as a "go-between", as any errors are more likely to stand out. The conquering of number systems opens many doors for future discussion including binary addition and subtraction, binary multiplication and division, logic elements, arithmetic logic units and the whole principle behind digital electronics.

☆☆☆

New Members

D.H. Ackland, Miranda
Craig Armsworth, Bilgola Plateau
Quy Bao, Newtown
Ken Bosward, Lugarno
R.J. Carey, Padstow
Graham Cook, Penshurst
Francis Curnow, Bankstown
Paul Dierking, Maroubra
P.J. Egan, Glen Innes
Brian Halligan, Mount Colah
Lindsay Heron, Speers Point
Russell Holmes, Eleebana
Gregory Hudson, Balgowlah Heights
Arthur Manrique, Kogarah
K.A. Nacinovich, Mona Vale
Peter Oldham, Dampier, W.A.
R.K. Onslow, Millfield
R.A. Penney, Winston Hills
Allan Percy, Redhead
Karl Pirchmoser, Tenterfield
C.V. Rosser, Alderley, Qld.
R. & D. Scott, Umina
Brian Simmons, Tennyson
Carey Smith, Santo, Vanuatu, New Hebrides
Brian Soutter, Davidson
Ian Steer, Emu Plains
Chris Townsend, East Sydney

INTRODUCTION TO FORTH

BY JOHN MATTES

The end of November, 1982 was a pretty exciting time; we took delivery of our new disk drive and Extended Fig-FORTH. Since then I have had a great deal of fun using FORTH and in this article I hope to share some of my (brief) experience with you.

Let me say from the start that I believe FORTH is far superior to basic in almost every respect. I will admit that it is a little harder to learn but well worth the effort.

A few things to know about FORTH. It works by defining WORDS, each of which extends the language and is therefore immediately available for use in programs other than the current one. FORTH does this by having FORTH words (mainly) operate on a STACK. So before you run a subroutine (that is invoke a FORTH word by simply typing in its name) you must first of all load the numbers it needs on the stack. To do this, you can type in the numbers, separated by blanks. FORTH recognises them as numbers and automatically puts them on the stack.

To give an example, the equivalent of PRINT 2+3 in basic is 2 3 + . in FORTH. The FORTH compiler recognises the 2 space (spaces are always significant in FORTH) and puts 2 on the stack. Similarly for the 3. Then it encounters the +, which is a FORTH word meaning "add the two top-most words on the stack (deleting them) and put the result back on the stack". The "." is the FORTH word for PRINT. Words can be (almost) any combination of symbols and letters up to 31 characters long.

One of the first exercises I did was to re-code ESCAPE (which was published in INSIDE INFO No. 3 and is also reproduced here). The result appears as Screens 37 and 38 (FORTH programs are stored as "screens"). I shall use this program as an example of FORTH and as a basis of comparison with BASIC.

Line 0 of Screen 37 is a comment line. Comments are enclosed in brackets. Lines 1, 2 and 3 define 5 variables, X1, X2, Y1, Y2 and C, giving the first four the initial value of zero and C the value of 1. Notice that named variables must be explicitly declared. The X and Y variables are used to hold the co-ordinater of PLOT and DRAW (TO) and C is used to hold the colour register number to be used when plotting or drawing.

Line 4 commences with a ":" which tells the compiler that a new word is going to be defined, in this case a word called BOX. As its name suggests, BOX draws a rectangle through X1, Y1 and X2, Y2. It does this be doing one PLOT at X1, Y1 and four DRAWs. These words require 3 numbers to be on the stack, the first specifying the colour register to be used for the point (or line) and the other two specifying the X and Y co-ordinates. In FORTH invoking the name of a variable (e.g. "C") causes the address of that variable to be placed on the stack. If we want the value of the variable we must fetch it from the address specified. This is done with the word "@" (called "fetch"). So the sequence "C @ X1 @ Y1 @" retrieves the contents of the variables C, X1 and Y1 and places them on the stack ready for PLOT (or DRAW in line 8) to operate on. The ";" in line 8 tells the compiler that the definition of BOX is finished.

Notice that BOX is perfectly general. Once compiled into your system you can draw a rectangle simply by giving the variables C, X1, Y1, X2 and Y2 values and saying "BOX"!

The word SETCR defined at line 10 sets the hue and luminance of 8 of the colour registers used in graphics mode 10. First of all, the 8 values are loaded onto the stack (in reverse order to that shown in the original basic program, since the last value on is the first off). A DO loop is started with an initial value of 1 and a final value of 9. Inside the loop, a number is pulled of the stack (2 on the first pass) added to 224 (to give a hue of 14 = orange-green and a luminance of 2). This is stored into byte 704 + I via the 704 I + C!. Inside a loop I is special FORTH word which puts the loop index on the stack and C! stores the 8 bit byte second last on the stack as the address specified by the last entry. On the first pass of the loop, I = 1 so 224 is stored at address 705, which is the colour register for Player 1. The word LOOP increments the loop index and if the result is greater than or equal to the limit (9 in this case) the loop terminates. So we need a limit of 9 to achieve a final pass of the loop with I = 8.

The word ROTATE On line 12 simply rotates the contents of memory locations between two specified limits (which must be on the stack). This word performs the same function (but in a more general way) as the machine códed routine called up in the USR statements in the basic version of ESCAPE. FORTH is so much faster than basic that machine code in unnecessary in this case (but note that most FORTH systems have their own assembler!). I will let you figure out how rotate works. The words not already described are:

DUP n1 --- n1 n1
Duplicate the last entry on the stack (symbols to the left of --- must be on the stack before invoking the word, symbols on the right are left by the word)

ROT n1 n2 n3 --- n2 n3 n1
Rotate the last 3 entries

OVER n1 n2 --- n1 n2 n1
Copy the second stack value making it the new top

- n1 n2 --- diff
Subtract

SWAP n1 n2 --- n2 n1

! n addr ---
Store the top (16 bit stack entry at addr n addr ---

Screen 37 ends on line 15 with a FORTH word which means "load the next screen". Line 1 starts the definition of a word RUN, which starts you running through the escape tunnel. RUN starts with a BEGIN statement (repetive execution), rotates colour registers 705 to 712 and executes an empty loop 600 times. Then UNTIL checks to see whether 2 = 0. It finds it is isn't (surprise!) and loops back to begin. UNTIL only terminates its loop if the preceeding test is TRUE, so the loop executes "forever". Note that the basic program (line 90) only needs 24 passes through an empty loop to slow the motion down to roughly the same speed. You can do "nothing" about 30 times faster in FORTH!

Finally we get to the ESCAPE program itself. In line 4, Graphics mode 10 is set and SETCR is invoked. Then a DO loop is executed 38 times to draw 38 pairs of BOXES. Two boxes are needed for each pass of the loop becuase Graphics mode 10 pixel is 4 elements wide by one high, giving 80 pixels across the screen and 192 down. So to get a reasonably symmetrical display we need two horizontal lines for each vertical line. This is accomplished (in this case) by the slightly inefficient tactic of drawing two overlapping boxes, the second having the same "walls" (same X1 and X2) but a lower "ceiling" and higher "floor" (Y1 = Y1 + 1, or 1 Y1 +! in FORTH and 1 MINUS Y2 +!).

After drawing the second box of the pair, C (which specifies the colour register to be used in drawing the next two boxes) is incremented by 1 and checked to see whether it exceeds 8. In this case, C is reset to 1, to restart the luminance sequence beginning with luminance = 2. After the 38 pairs of boxes have been drawn, the loop terminates and you start to RUN through the tunnel.

To actually run ESCAPE you have to LOAD screen 37 by typing 37 LOAD. This compiles all the words in Screens 37 and 38. Then you simply type in ESCAPE.

Finally, FORTH is much more compact than BASIC. Escape takes up 756 bytes in BASIC but only 396 bytes in FORTH. So you can see FORTH is a language with a lot going for it!

```
SCR # 37
  0 (      DEMO PROGRAM JEM-10DEC82 )
  1      0 VARIABLE X1  0 VARIABLE Y1
  2      0 VARIABLE X2  0 VARIABLE Y2
  3      1 VARIABLE C
  4 : BOX C @ X1 @ Y1 @ PLOT
  5       C @ X2 @ Y1 @ DRAW
  6       C @ X2 @ Y2 @ DRAW
  7       C @ X1 @ Y2 @ DRAW
  8       C @ X1 @ Y1 @ DRAW ;
  9 : SETCR 0 2 4 6 8 6 4 2  9 1 DO
 10   224 + 704 I + C! LOOP          ;
 11
 12 : ROTATE DUP @ ROT ROT OVER
 13     SWAP DO I 1+ @ I C! LOOP C! ;
 14
 15 --

SCR # 38
  0
  1 : RUN BEGIN 712 705 ROTATE
  2     600 1 DO LOOP 2 0= UNTIL ;
  3
  4 : ESCAPE 10 GR. SETCR
  5   39 0 DO
  6           I X1 !      I I + Y1 !
  7    79 X1 @ - X2 ! 190 Y1 @ - Y2 !
  8    BOX
  9    1 Y1 +!   1 MINUS Y2 +!   BOX
 10    1 C +! C @ 8   IF 1 C ! ENDIF
 11   LOOP  RUN  ;
 12
 13
 14
 15
```

BOXES

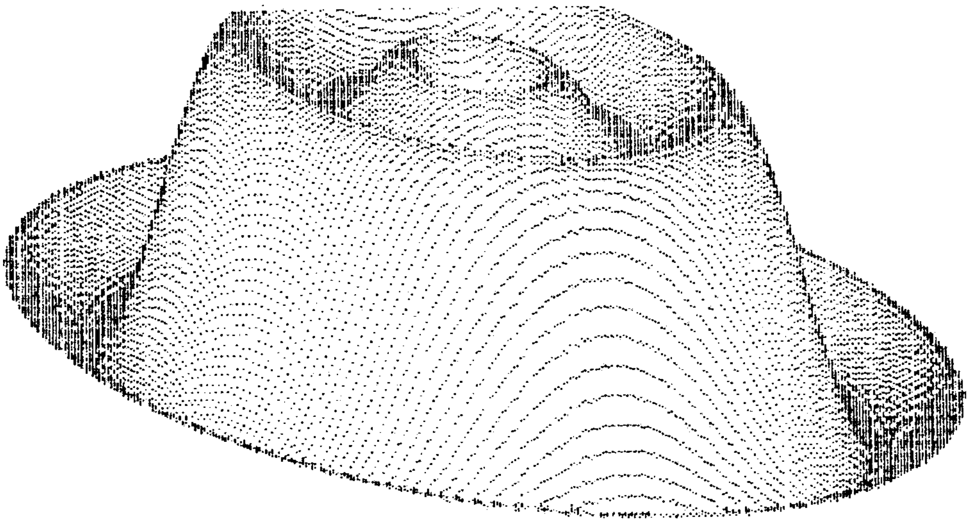
by John Trigge

Here's a small game that I've been working on. It is based on the game of boxes that kids play at school. The game is a fairly well known one, so the instructions provided in the program should be sufficient to enable play without any explanation – at least that was my intention whilst writing the program. I have tested it on two or three people and they all appeared to manage okay.

```
1 REM ###############################
2 REM #            BOXES            #
3 REM #       by John Trigge        #
4 REM # Published by Atari Computer #
5 REM #    Enthusiasts (N.S.W.)     #
6 REM #         April 1983          #
7 REM ###############################
10 GOSUB 890
20 GRAPHICS 17:C=8:? #6:? #6;" MOVE JO
YSTICK 1 UP"
30 ? #6;" OR DOWN TO CHOOSE"
40 ? #6;" GRID SIZE"
50 ? #6:? #6;" press trigger to"
60 ? #6;" start game"
70 ? #6:? #6;"    16 SQUARES"
80 ? #6:? #6;"    25 SQUARES"
90 ? #6:? #6;"    36 SQUARES"
100 ? #6:? #6;"    49 SQUARES"
110 ? #6:? #6;"    64 SQUARES"
120 ? #6:? #6;"    81 SQUARES"
130 ? #6:? #6;"    USER DEFINED"
140 IF STRIG(0)=0 THEN 140
150 POSITION 3,C:? #6;"≥":FOR W=1 TO 1
00:NEXT W
160 IF STICK(0)=13 THEN POSITION 3,C:?
 #6;" ":C=C+2:IF C=22 THEN C=8
170 IF STICK(0)=14 THEN POSITION 3,C:?
 #6;" ":C=C-2:IF C=6 THEN C=20
180 IF STRIG(0)=1 THEN 150
190 IF C<>20 THEN H=C/2:V=H:GOTO 340
200 GRAPHICS 17:? #6:? #6;" INPUT GRID
 SIZE BY"
210 ? #6;" MOVING JOYSTICK 1"
220 ? #6;" UP OR DOWN UNTIL"
230 ? #6;" DESIRED VALUE IS"
240 ? #6;" SHOWN"
250 ? #6:? #6;" press trigger to":? #6
;" enter value"
260 C=1:P=18:POSITION 4,P:? #6;"horizo
ntal":L=19:GOSUB 280:H=C
270 C=1:P=20:POSITION 6,P:? #6;"vertic
al":L=9:GOSUB 280:V=C:GOTO 340
280 IF STRIG(0)=0 THEN 280
290 POSITION 15,P:? #6;C;" ":FOR W=1 T
O 100:NEXT W
300 IF STICK(0)=14 THEN C=C+1:IF C>L T
HEN C=1
310 IF STICK(0)=13 THEN C=C-1:IF C<1 T
HEN C=L

320 IF STRIG(0)=1 THEN 290
330 RETURN
340 GRAPHICS 5:SETCOLOR 0,6,8:SETCOLOR
 1,12,12:SETCOLOR 2,1,4:COLOR 3:POKE 7
52,1
350 HOR=39-2*H:VER=19-2*V
360 FOR I=HOR TO HOR+H*4 STEP 4
370 FOR J=VER TO VER+V*4 STEP 4
380 PLOT I,J:NEXT J:NEXT I
390 DIM GRID(H,V),SC(1):FOR I=1 TO H:F
OR J=1 TO V:GRID(I,J)=1:NEXT J:NEXT I:
SC(0)=0:SC(1)=0
400 PL=0:HPOS=HOR:VPOS=VER
410 IF STRIG(0)=0 THEN 410
420 ? CHR$(125);"PLAYER ";PL+1;":":? "
USE JOYSTICK ";PL+1;" TO SELECT STARTI
NG":? "POINT, THEN PRESS TRIGGER"
430 COLOR PL+1:PLOT HPOS,VPOS:FOR W=1
TO 50:NEXT W:SOUND 0,0,0,0
440 IF STRIG(PL)=0 THEN 510
450 IF STICK(PL)=14 THEN GOSUB 500:VPO
S=VPOS-4:IF VPOS<VER THEN VPOS=VER+4*V
460 IF STICK(PL)=13 THEN GOSUB 500:VPO
S=VPOS+4:IF VPOS>VER+4*V THEN VPOS=VER
470 IF STICK(PL)=11 THEN GOSUB 500:HPO
S=HPOS-4:IF HPOS<HOR THEN HPOS=HOR+4*H
480 IF STICK(PL)=7 THEN GOSUB 500:HPOS
=HPOS+4:IF HPOS>HOR+4*H THEN HPOS=HOR
490 COLOR 3:PLOT HPOS,VPOS:FOR W=1 TO
50:NEXT W:GOTO 430
500 POKE 77,0:COLOR 3:PLOT HPOS,VPOS:S
OUND 0,40+PL*40,10,6:RETURN
510 POKE 77,0:? CHR$(125);"PLAYER ";PL
+1;":":? "MOVE JOYSTICK RIGHT OR DOWN
TO DRAW":? "LINE"
520 IF STICK(PL)=13 THEN S=1:GOTO 550
530 IF STICK(PL)=7 THEN S=2:GOTO 550
```

```
540 GOTO 520
550 POKE 77,0:HP=(HPOS-HOR)/4+1:VP=(VP
OS-VER)/4+1
560 IF ((HP=H+1) AND STICK(PL)=7) OR (
(VP=V+1) AND STICK(PL)=13) THEN 580
570 ON S GOTO 610,680
580 ? CHR$(125):? "ILLEGAL MOVE"
590 FOR I=1 TO 5:SOUND 0,200,10,10:FOR
 J=1 TO 20:NEXT J:SOUND 0,100,10,10:FO
R J=1 TO 20:NEXT J:NEXT I
600 GOTO 420
610 IF HP>1 THEN IF GRID(HP-1,VP)/3=IN
T(GRID(HP-1,VP)/3) THEN 580
620 IF HP<=H THEN IF GRID(HP,VP)/7=INT
(GRID(HP,VP)/7) THEN 580
630 PLOT HOR+4*(HP-1),VER+4*(VP-1)+1:D
RAWTO HOR+4*(HP-1),VER+4*(VP-1)+3
640 IF HP>1 THEN GRID(HP-1,VP)=GRID(HP
-1,VP)*3:IF GRID(HP-1,VP)=210 THEN GOS
UB 740:GOSUB 780:FL=1
650 IF HP<=H THEN GRID(HP,VP)=GRID(HP,
VP)*7:IF GRID(HP,VP)=210 THEN GOSUB 75
0:GOSUB 780:FL=1
660 IF FL=0 THEN PL=1-PL
670 FL=0:GOTO 420
680 IF VP>1 THEN IF GRID(HP,VP-1)/5=IN
T(GRID(HP,VP-1)/5) THEN 580
690 IF VP<=V THEN IF GRID(HP,VP)/2=INT
(GRID(HP,VP)/2) THEN 580
700 PLOT HOR+4*(HP-1)+1,VER+4*(VP-1):D
RAWTO HOR+4*(HP-1)+3,VER+4*(VP-1)
710 IF VP>1 THEN GRID(HP,VP-1)=GRID(HP
,VP-1)*5:IF GRID(HP,VP-1)=210 THEN GOS
UB 760:GOSUB 780:FL=1
720 IF VP<=V THEN GRID(HP,VP)=GRID(HP,
VP)*2:IF GRID(HP,VP)=210 THEN GOSUB 77
0:GOSUB 780:FL=1
730 GOTO 660
740 FOR I=1 TO 3:PLOT HOR+4*(HP-2)+I,V
ER+4*(VP-1)+1:DRAWTO HOR+4*(HP-2)+I,VE
R+4*(VP-1)+3:NEXT I:RETURN
750 FOR I=1 TO 3:PLOT HOR+4*(HP-1)+I,V
ER+4*(VP-1)+1:DRAWTO HOR+4*(HP-1)+I,VE
R+4*(VP-1)+3:NEXT I:RETURN
760 FOR I=1 TO 3:PLOT HOR+4*(HP-1)+I,V
ER+4*(VP-2)+1:DRAWTO HOR+4*(HP-1)+I,VE
R+4*(VP-2)+3:NEXT I:RETURN
770 FOR I=1 TO 3:PLOT HOR+4*(HP-1)+I,V
ER+4*(VP-1)+1:DRAWTO HOR+4*(HP-1)+I,VE
R+4*(VP-1)+3:NEXT I:RETURN
780 SC(PL)=SC(PL)+1
790 IF SC(0)+SC(1)<H*V THEN RETURN
800 FOR I=1 TO 10:FOR J=1 TO 15
810 SETCOLOR 4,J,4:SOUND 0,J*10,10,J
820 NEXT J:NEXT I
830 SOUND 0,0,0,0
840 ? CHR$(125);:POKE 657,17:? "SCORES
":POKE 657,14:? "PLAYER 1...";SC(0)
850 POKE 657,14:? "PLAYER 2...";SC(1)
860 ? "      PRESS TRIGGER TO PLAY AGAI
N";
870 IF STRIG(0)=0 OR STRIG(1)=0 THEN C
LR :GOTO 20
880 GOTO 870
890 GRAPHICS 0:POKE 752,1:POSITION 17,
1:? "BOXES"
900 ? :? "Two players take turns to co
nnect":? "adjoining points on a grid.
 Each"
910 ? "time a player completes a squar
e, he":? "scores 1 point and gets an e
xtra"
920 ? "turn.  The winner is the player
 with":? "the most points when all the
 squares":? "are completed."
930 ? :? "In this version of the game,
 lines":? "can only be drawn from left
 to right":? "or top to bottom."
940 POSITION 5,22:? "PRESS EITHER TRIG
GER TO BEGIN"
950 IF STRIG(0)=0 OR STRIG(1)=0 THEN R
ETURN
960 GOTO 950
```



GREEK ALPHABET

by Jamie Athas

The ATARI computer can do just about anything. [I haven't seen it do the washing up...yet! - GF] Wouldn't it be great if it could write in different languages? This program presents a routine which allows you to print the Greek alphabet. In its present form, you merely press any alphabetic key to print the normal English character, but by pressing the CONTROL key as well, you can print the equivalent Greek character. In actual fact, the program goes one step <u>too</u> far. There are 24 letters in the Greek alphabet, but 26 characters have been redefined. This is because Y and Z have been used for accents.

The program uses the machine language subroutine from DOGGIES to move the character set (lines 20 to 40). This means that initialisation only takes 3 seconds. Note that pressing SYSTEM RESET or doing a GRAPHICS command will reset the pointer to the character set and the Greek characters may be lost.

[Let's see some more foreign character fonts from other members. How about German, Russian, Arabic and Japanese for starters? And how about some applications for them? - GF]

```
1 REM ##############################
2 REM #       GREEK ALPHABET       #
3 REM #        by Jamie Athas      #
4 REM # Published by Atari Computer #
5 REM #    Enthusiasts (N.S.W.)    #
6 REM #         April 1983         #
7 REM ##############################
10 GRAPHICS 0:POKE 559,0:POKE 712,12:P
OKE 710,12:POKE 709,4:START=PEEK(106)-
8
20 DIM ML$(32):FOR I=1 TO 32:READ A:ML
$(I)=CHR$(A):NEXT I:CHSET=256*START:X=
USR(ADR(ML$),57344,CHSET)
30 DATA 104,104,133,204,104,133,203,10
4,133,206,104,133,205,162,4
40 DATA 160,0,177,203,145,205,136,208,
249,230,204,230,206,202,208,240,96
50 FOR I=CHSET+520 TO CHSET+728:READ A
:POKE I,A:NEXT I:POKE 756,START
60 DATA 0,24,60,102,102,126,102,102,0,
124,102,102,124,102,102,124
70 DATA 0,127,50,48,48,48,48,120,0,24,
60,102,102,102,102,126
80 DATA 0,126,50,48,60,48,50,126,0,126
,70,12,24,48,98,126
90 DATA 0,102,102,126,126,102,102,102,
0,60,102,102,126,102,102,60
100 DATA 0,126,24,24,24,24,24,126,0,23
0,102,108,120,120,108,102
110 DATA 0,24,60,102,102,102,102,231,0
,99,127,107,99,99,99,99
120 DATA 0,102,102,118,126,110,102,102
,0,126,0,0,60,0,0,126
130 DATA 0,60,102,102,102,102,60,0,0,2
55,102,102,102,102,102,231
140 DATA 0,252,102,102,124,96,96,240,0
,126,98,48,24,48,98,126
150 DATA 0,126,24,24,24,24,24,24,0,195
,102,60,24,24,24,24
160 DATA 0,8,62,107,107,107,62,8,0,102
,102,24,24,102,102,195
170 DATA 0,8,107,107,107,62,8,56,0,60,
102,102,102,60,0,126
180 DATA 0,0,0,0,0,6,12,24,0,0,0,0,0,0
,62,0,0
190 POKE 82,13:? CHR$(125);"GREEK ALPH
ABET":?
200 FOR I=65 TO 77:? CHR$(I);" = ";CHR
$(I-64);"    ";CHR$(I+13);" = ";CHR$(I
-51):NEXT I
210 POKE 82,2:? :? "PRESS CTRL AND ENG
LISH CHARACTER TO":? "GET EQUIVALENT G
REEK CHARACTER."
220 POKE 559,34:END
```

* * *

ROSE

```
1 REM ##############################
2 REM #            ROSE            #
3 REM #       by Greg Marthick     #
4 REM # Published by ATARI Computer #
5 REM #    Enthusiasts (N.S.W.)    #
6 REM #         April 1983         #
7 REM ##############################
10 GRAPHICS 22
20 SETCOLOR 4,INT(16*RND(0)),INT(4*RND
(0))
30 SETCOLOR 0,INT(16*RND(0)),INT(12*RN
D(0))+4
40 M=30*RND(0)+3:N=M*RND(0)+1:MN=M/N
50 X=79:Y=47:COLOR 1:PLOT X,Y
60 FOR I=0 TO 128 STEP RND(0)+0.1
70 R=Y*SIN(MN*I)
80 DRAWTO X+R*SIN(I),Y+R*COS(I)
90 SOUND INT(3*RND(0)),INT(200*RND(0))
+56,10,INT(16*RND(0))
100 NEXT I:RUN
```

Odds & Sods

A regular column by Garry Francis

All members who joined before 1st June, 1982 will have their membership extended to 1st June, 1983. This ensures that they will receive the full complement of 6 magazines along with everybody else. Please note that there will be no freebies. Anybody who does not renew their subscription when their time comes will be taken off the mailing list.

* * *

The second batch of blank discs are almost sold out. We have come up with a new supplier, but the price will go up by about $7 and there will be no plastic box supplied. Buy now to get them at the old price of $28 per box of 10. First in, first served.

* * *

As mentioned last issue, the Software Exchange is now operational. The first disc is now available. This is a utilities disc drawn mainly from ANTIC's Public Domain disc. The programs include an automatic line number generator, memory tester, 2 assemblers, documentation program, parallel printer driver for Ports 3 & 4 and a program to make disc housecleaning a simpler task. All documentation is included on the disc. By the time you read this, the second disc should also be available. This will be all games. The third disc should be available shortly. This is an extended version of fig-FORTH from Team ATARI. It comes with full graphics, sound and player-missile graphics support. At $10, it's the best bargain you'll get this century! You'll also need a book or two on FORTH and the documentation for this version out of SoftSide #36. This must be bought separately (for copyright reasons) and is available now at selected newsagents.

* * *

Overseas subscriptions will now be $25 per year to help absorb the exhorbitant airmail postage rates.

* * *

We were approached by a few people at the 1st Australian Personal Computer Show about starting a group in Newcastle. Anybody interested?

* * *

There's a new club which may be of interest to our South Australian members (if we've got any). It's called the Adelaide Atari Computer Club. They meet at Sargent Electronics, 6 Moseley Street, Glenelg, S.A. 5045. They have a newsletter (which includes stuff from INSIDE INFO - boy, it sure gets around!) and the usual user group activities. Write to the above address for details.

* * *

That's all for now. Please don't forget to vote (see Editorial for details). Next time, I'll be back with some very interesting hints and tips. Happy computing.

COMPUTERWAVE

WILL SOON MOVE TO

BIGGER PREMISES

NEAR BY MYER – GRACE BROTHERS

WATCH FOR ANNOUNCEMENT

WE HAVE A HUGE NEW RANGE OF ATARI SOFTWARE TO SHOW YOU

WE WILL NOT BE UNDERSOLD & WILL MEET ANY ADVERTISED PRICE

WE STOCK ALL IMAGINEERING SOFTWARE

WE WILL SUPPLY MAIL ORDER

WE ACCEPT BANKCARD

COMPUTERWAVE
PTY LIMITED

All Correspondence to:
Box 268
GPO Sydney
N.S.W. 2001

Lower Ground Floor
Myer Sydney Store
George and Market Streets
Sydney
Tel: 238 9984